One-Hour CoasterBot

Written By: Matt Mets

🔧 TOOLS:

- 5v FTDI programming cable (1)
- Needle Nose Pliers (1)
- Scissors (1)
- Soldering Iron and rosin core solder. (1)
- Wire cutter/stripper (1)

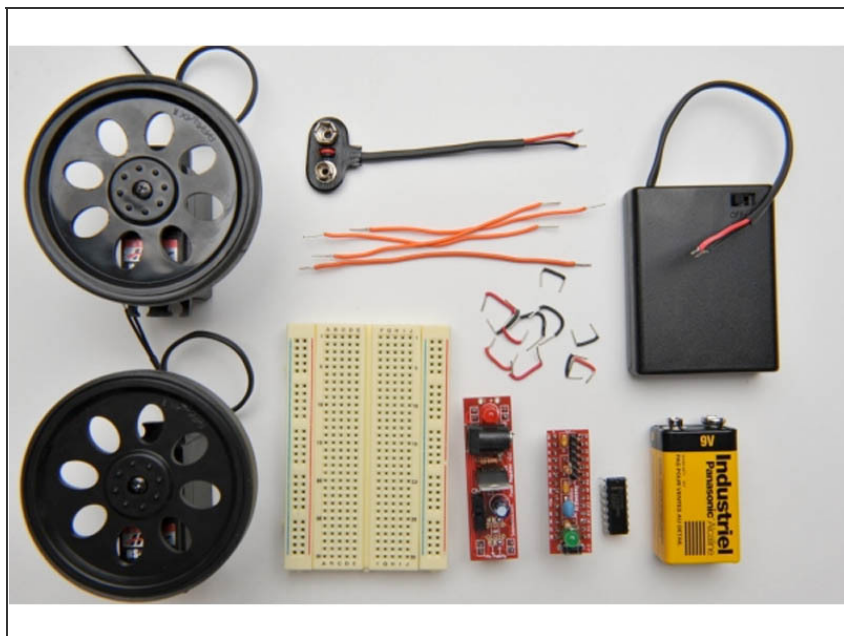### ⚙ PARTS:

- Servo (generic) (2)
- 9v battery snap (1)
- 9 Volt alkaline battery (1)
- Mini breadboard (1)
- Ardweeny microcontroller (1)
- Breadboard Power Regulator (1) *from Solarbotics*
- L293DNE H-bridge (1)
- 22 AWG, solid core wire (1)
- Double-sided tape (1)
- Dead CDs/DVDs (2)
- Suitable robot wheels (2)

SUMMARY

Last year we ran a cool CoasterBot build project and contest on MAKE. We had dozens of awesome entries, from the most basic to the quite elaborate. This Make: Project shows you how to make just about the simplest CoasterBot imaginable. From here you can add new sensors and features and build on the basic bot-building knowledge you learn in this project.

Step 1 — Gather Your Parts

- You can either gather the parts yourself or buy the Jameco CoasterBot parts bundle that we put together with them for our CoasterBot build project. It comes with all of the major parts you need (including the Ardweeny, the Breadboard Regulator, the servos, wheels, battery connector and battery snap, wire, etc.). See the Jameco page above for complete parts list.

- If you'd like to buy just the Ardweeny and Power Regulator, you can get the Ardweeny here and the Power Regulator here.

- These instructions assume that you've already assembled the Ardweeny and the Solarbotics Power Regulator based on the instructions that came with them.

- You also need to hack your servomotors for continuous rotation. One of the CoasterBot contestants created this handy how-to.

## Step 2 — Understanding the Circuit on the Breadboard
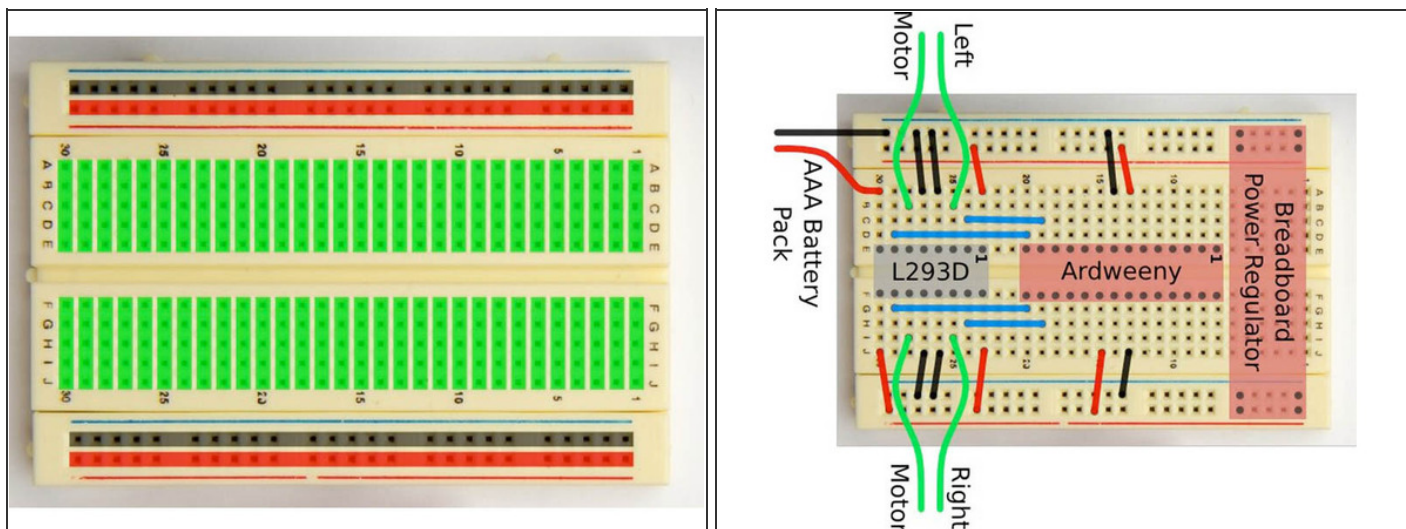
- A breadboard acts as as big interconnect for the electrical parts, allowing you to easily hook up power and other control signals without running too many wires. It is divided up into three sections: two side rails for power, and one main section for connecting up components.

- Here, we have highlighted how the holes are connected together. Each shaded area represents a separate circuit, and all of the holes in that circuit are shorted together. Each one of these sections can be used to make a single connection, or 'node' in a circuit diagram. The two side sections are reserved for power — anything plugged into the black areas becomes connected to ground (-), and anything plugged into the red areas becomes connected to power (+). For a more on breadboards, see Tom Igoe's breadboard page.

- The second pic shows the circuit we are going to make, with the exact placement of each component and wire on the breadboard.

Step 3 — The Breadboard Power Supply

- We'll use a 9V battery to power the Solarbotics Breadboard Power Regulator. To hook it up, solder the positive (red) wire to the metal tab on the back of the power jack (as shown above), and the negative (black) wire to the metal tab on the side.

- Plug in the 9V battery, and flip the switch on the Breadboard Power Regulator to the 'on' position. If you've got it right, the red LED on the board should light up. Flip the switch back off for now.

## Step 4 — Breadboarding the Core Components
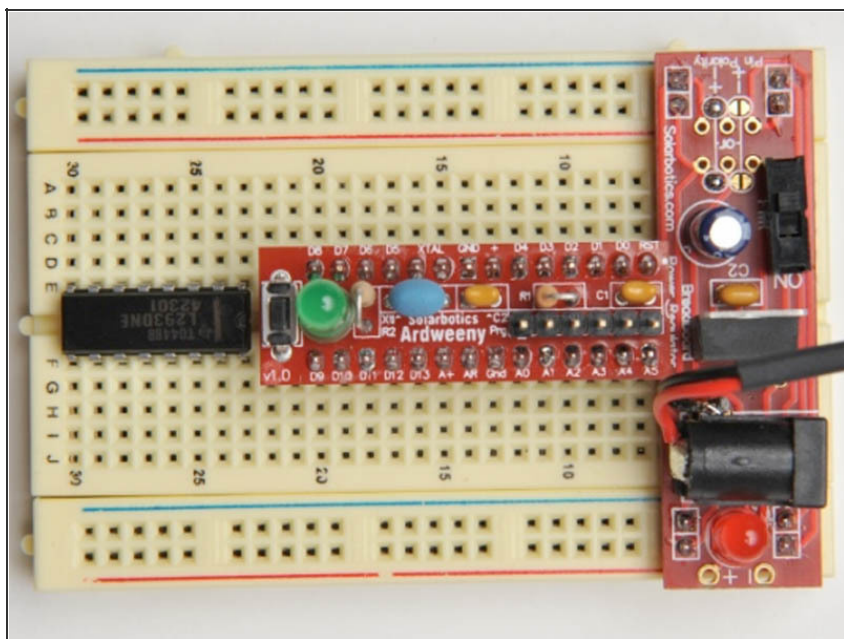
- Plug the basic components (power regulator, Ardweeny, and L293 motor driver) into the breadboard.

- It can be hard to tell exactly where to plug in everything by looking at the photo, so if you get confused, check out the layout diagram in Step 2.

Step 5 — Connecting Power Wires
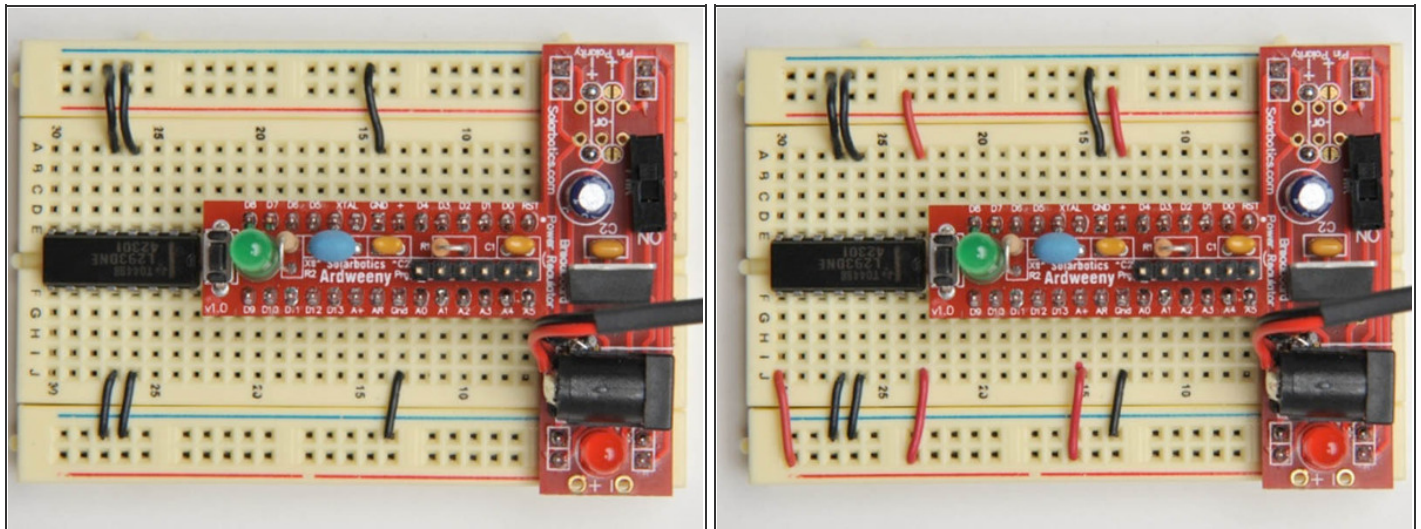
- Once the components are in place, the next step is to place jumper wires to ground any component leads that need grounding. These are shown in black.

- In a similar manner, use jumper wires to connect the pins that require 5v power. These are shown in red.

Step 6 — Adding Motor Control and Motors
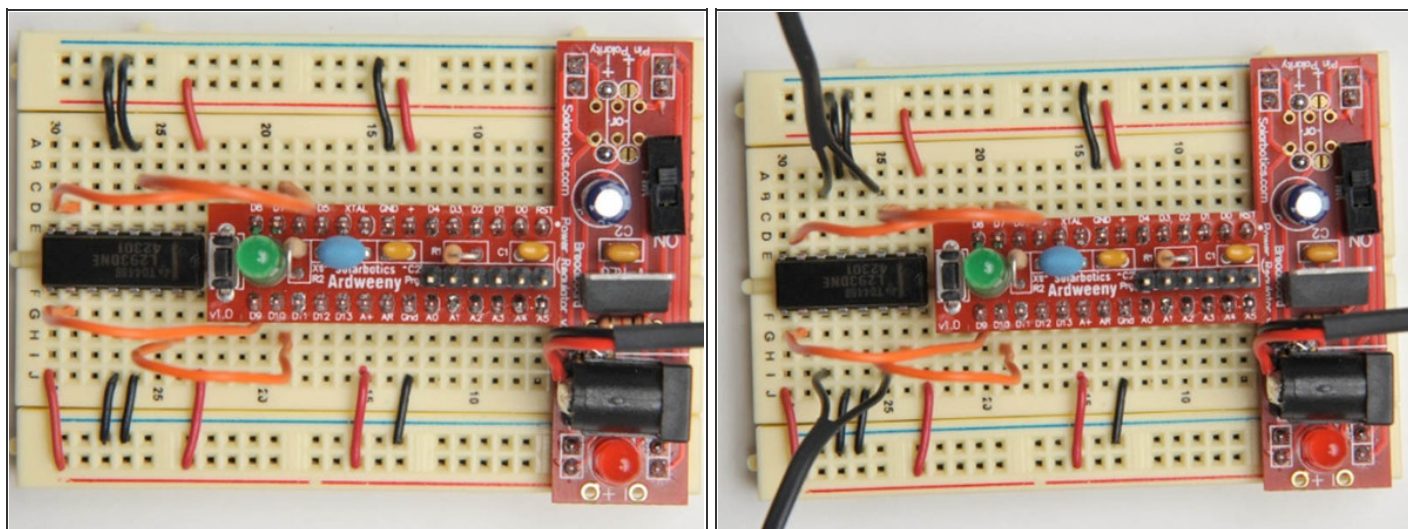
- The final set of jumper wires runs between the Ardweeny and the L293D, and allows the Ardweeny to control the motors. I used orange wires for my actual board, but they are blue in the drawing in Step 2.

- Now, you are ready to connect the leads from the continuous rotation servos to the L293D motor driver chip. Connect one motor to each side of the chip. We'll sort out which direction is which in the software later!

## Step 7 — Adding the Battery Power Pack
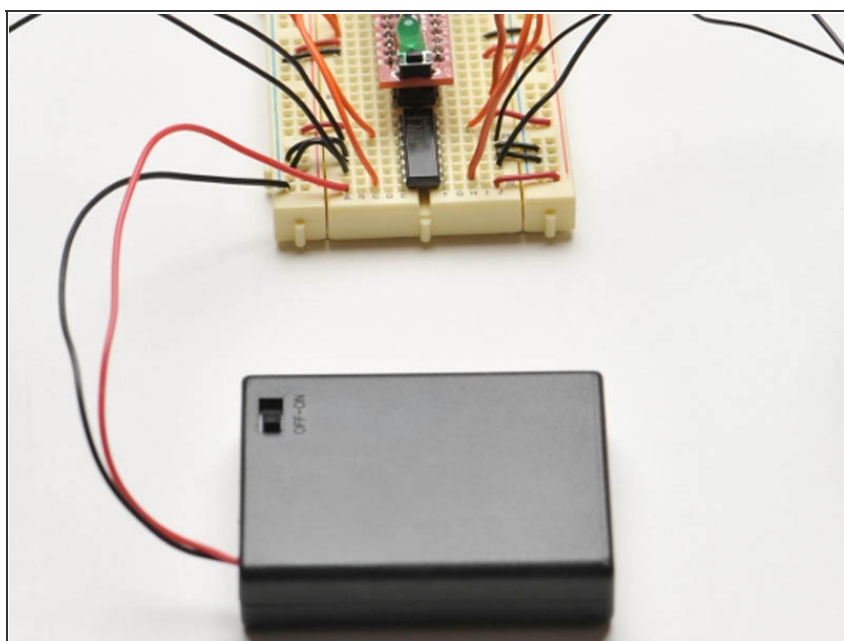
- The last thing to hook up is the AAA battery box, which will power the motors. This plugs directly into ground and Pin 8 pin on the L293D motor controller.

Step 8 — Downloading the Sample Code
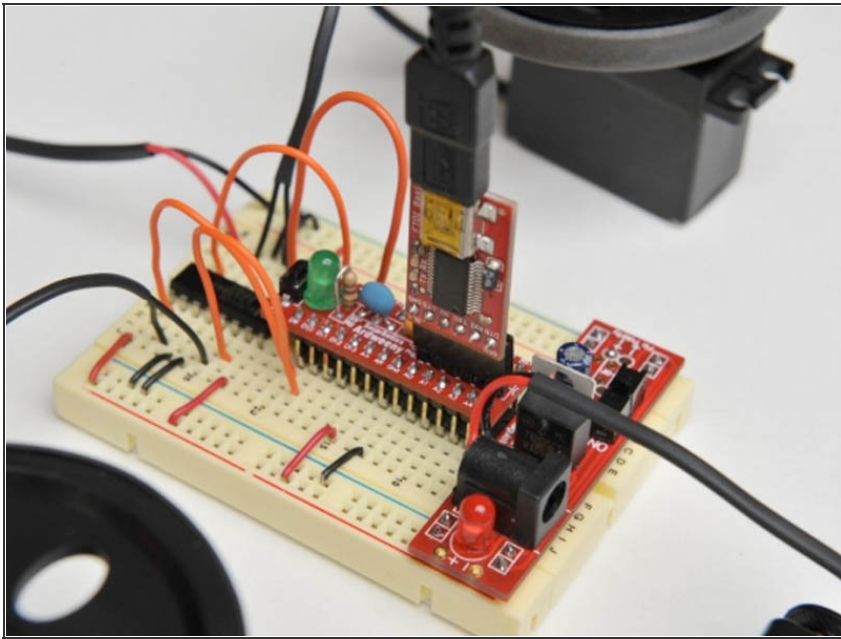
- All right! That's all the wiring you need to do to build a control system brain for your robot! The final step is to try downloading a program to the Ardweeny, and see if you can make the robot work!

- Make sure you have the Arduino software set up on your computer, then grab this sketch, and upload it to your bot. If everything is set up correctly, the motors should start to spin in a pattern!

Step 9 — Assembling the Chassis
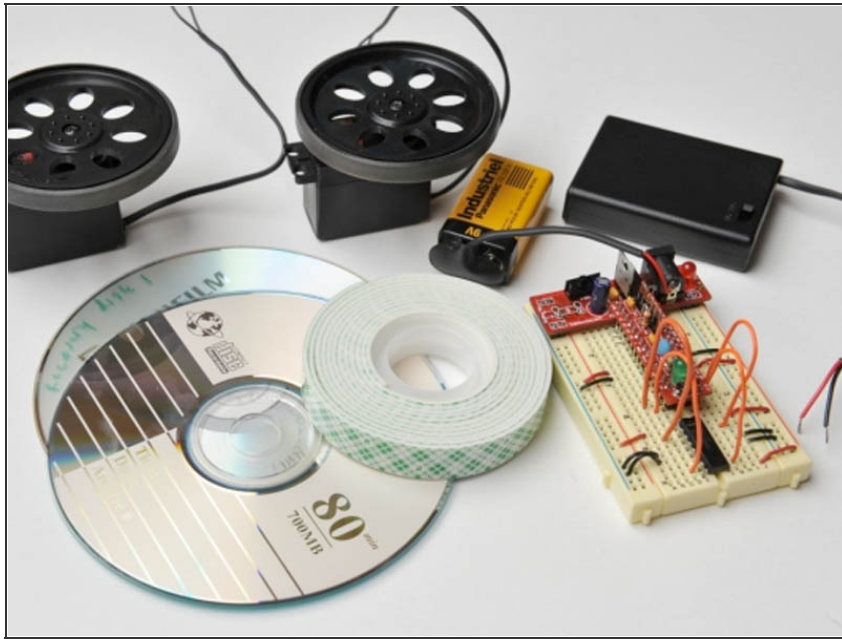
- Having a working control system is fun, but it's way cooler when it is actually attached to a chassis, so that it can start to move! To get you going as quickly as possible, we present our patented method for making a quick robot: double-sided tape! For this simple chassis, we'll skip the screws and glue, and use double-sided foam tape to patch everything together. It should work great for prototyping; however you will probably want to think of a more permanent solution. Once you do, just pull the components off of this temporary body and use them in your next chassis!

## Step 10 — Attaching the Servos to the Bottom CD
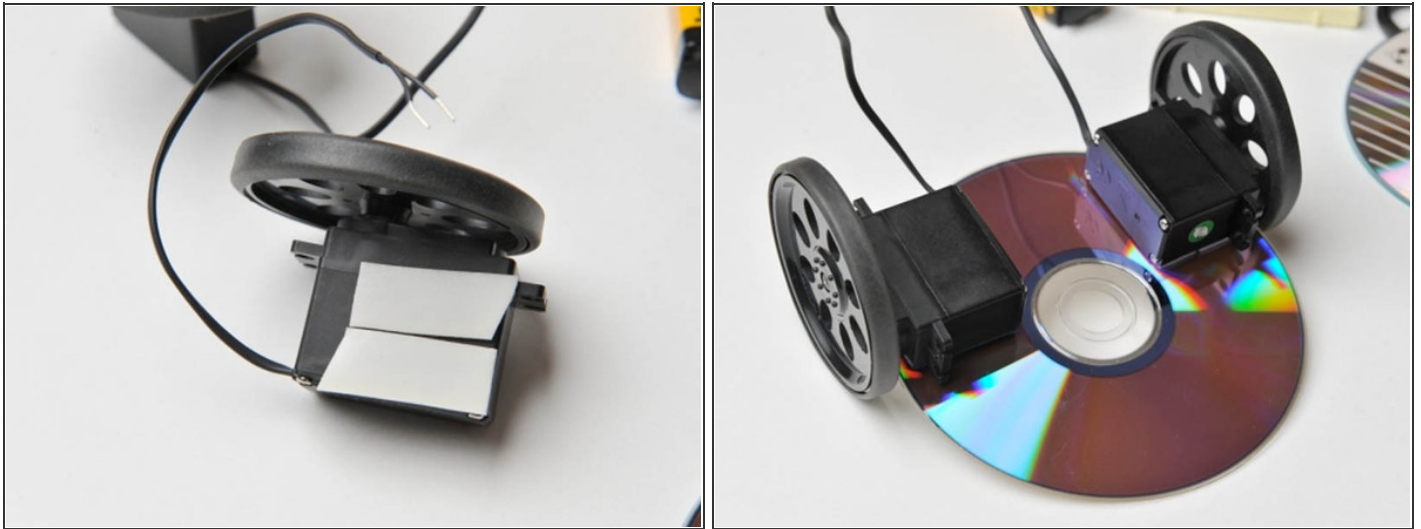
- Cut two pieces of sticky tape and stick them to one of the servo motors.

- Stick the servo near the edge of one of the CDs.

- Repeat for the second servo, making sure to leave enough space between them to fit the AAA battery pack.

Step 11 — Attaching the Top CD to the Servos
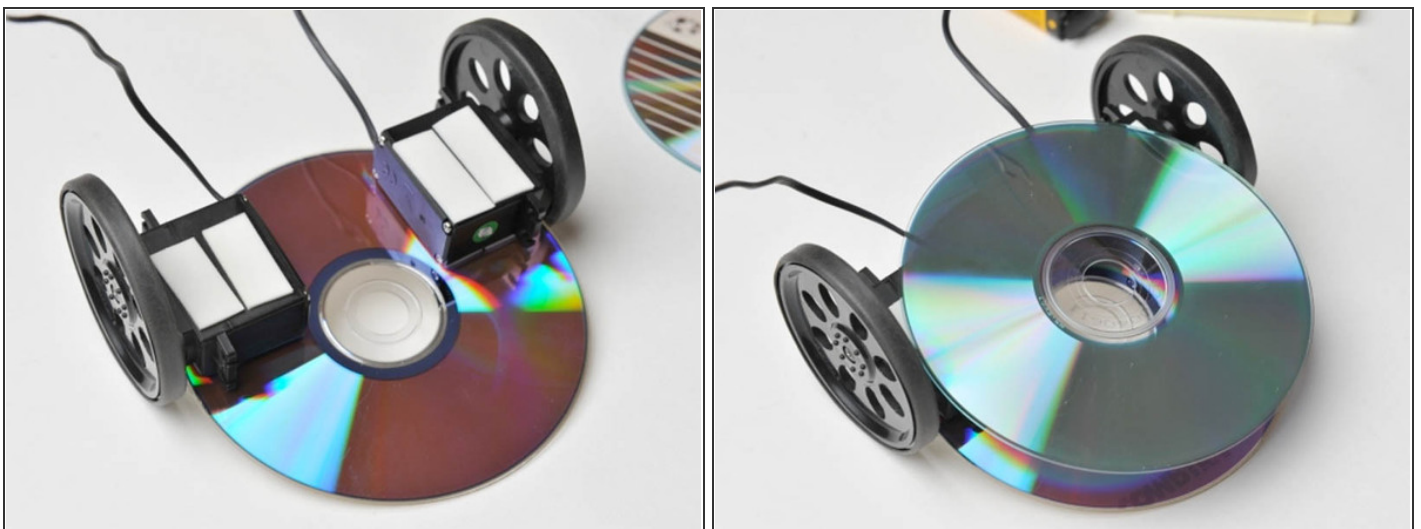
- Add some more tape to the top of both servo motors.

- Stick down the second CD, making sure to align it with the first.

Step 12 — Installing Breadboard and Battery Pack
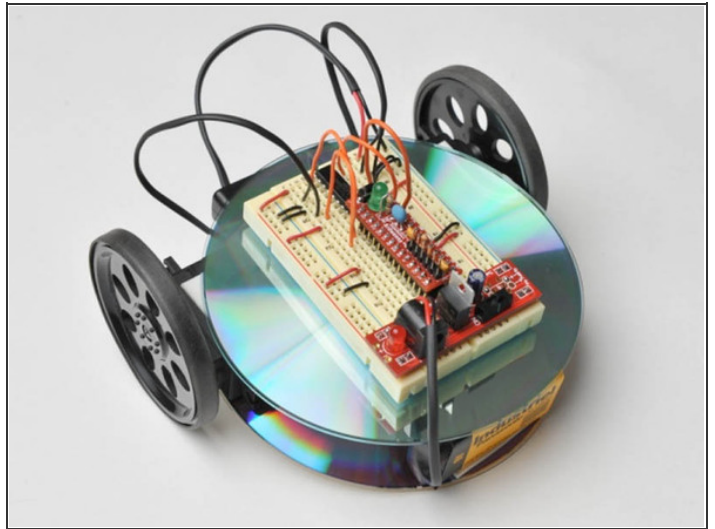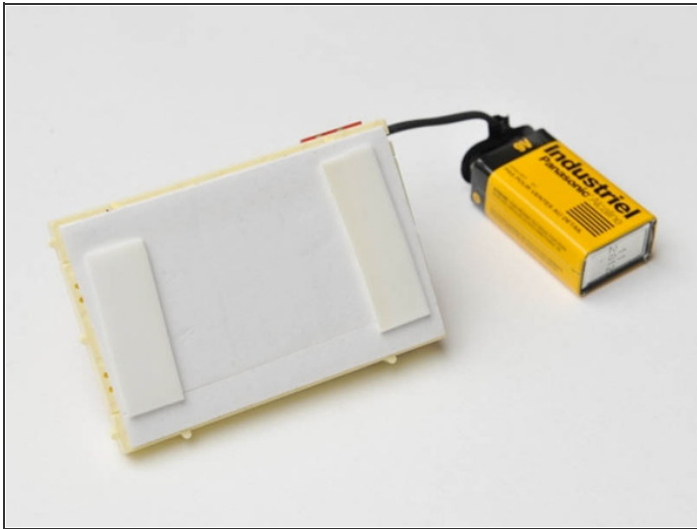
- Stick tape on back side of the breadboard

- Wedge in the 9V battery, using tape to hold it in place.

- That's it! if you plug the motors and battery pack in, the spinning motor test code from before should now let your robot do a little jig! We suggest you do one as well to celebrate!

If you've made it this far, you hopefully now have a functional robot body to start playing with. Next steps include shoring up the mechanical design (maybe use the spaces instead of tape?), adding sensors, and perfecting that code. Happy building!

This document was last generated on 2013-01-23 04:16:47 PM.